高等学校电子信息类“十二五”规划教材

C语言程序设计

主编　张丽君　李琳芳　赵欣

西安电子科技大学出版社
http://www.xduph.com

高等学校电子信息类“十二五”规划教材

C 语言程序设计

主　编　张丽君　李琳芳　赵　欣
副主编　张树静　赵晓莉　胡鹏飞　魏红娟　张海燕

西安电子科技大学出版社

内 容 简 介

本书共分为十三章，循序渐进地讲述了C语言的语法规则和编程思想，主要内容包括数据的存储和获取、屏幕的输入与输出、运算符、表达式、分支语句、循环语句、函数、数组、指针、字符串处理、结构体、共用体、枚举、位运算、文件处理、作用域、预处理、数据结构等知识点。每一章都配有一定数量的习题，供读者练习时使用。

本书可作为高等院校电子、通信和计算机等专业的基础课程的教材，也可作为程序开发人员的学习资料。

图书在版编目(CIP)数据

C语言程序设计/张丽君等主编.
—西安：西安电子科技大学出版社，2014.5
高等学校电子信息类“十二五”规划教材
ISBN 978-7-5606-3325-1

Ⅰ. ① C… Ⅱ. ① 张… Ⅲ. ① C语言—程序设计—高等学校—教材
Ⅳ. ① TP312

中国版本图书馆CIP数据核字(2014)第065272号

策　　划　毛红兵
责任编辑　毛红兵　董柏娴
出版发行　西安电子科技大学出版社（西安市太白南路2号）
电　　话　(029)88242885　88201467　邮　　编　710071
网　　址　www.xduph.com　电子邮箱　xdupfxb001@163.com
经　　销　新华书店
印刷单位　陕西天意印务有限责任公司
版　　次　2014年5月第1版　2014年5月第1次印刷
开　　本　787毫米×1092毫米　1/16　印张20.5
字　　数　487千字
印　　数　1～3000册
定　　价　35.00元
ISBN 978－7－5606－3325－1／TP
XDUP 3617001-1

前　言

本书依据国家计算机等级考试的要求安排内容，以 VC++6.0 为编程环境，向读者介绍 C 语言的基本语法及编程技巧。本书既适合程序设计语言的初学者入门学习之用，也可作为高等院校电子、通信和计算机专业的教材，还可作为培训机构的培训资料和开发人员的参考手册。

本书对每个知识点的讲解分为两部分：(1) 基础知识的讲解，主要对 C 语言的基本语法和结构等进行讲解；(2) 程序举例与运行测试，在对基础知识进行讲解的基础上，列举很多实例，加深读者对基础知识的理解，同时将程序的运行结果直接置于例题后面，读者可以不借助于计算机就能纠正对程序理解的错误之处，很大程度提高了读者的学习效率。

本书共分十三章：第 1 章概述，主要讲解 C 程序的特点及 VC++6.0 编程环境；第 2 章算法，主要讲解算法的基本思想以及常见的算法的表示方法等；第 3 章基本数据类型，主要讲解 C 语言的基本数据类型(包括整型、实型和字符型)及常用的运算(包括算术运算、关系运算和逻辑运算)；第 4 章顺序结构程序设计，主要讲解顺序结构程序的设计思想以及输入与输出语句的使用；第 5 章选择结构程序设计，主要讲解选择结构的程序设计思想，关系运算和逻辑运算等的运算规则，if 语句的基本格式，switch 语句的基本格式；第 6 章循环控制，主要介绍循环结构的程序设计思想，while 循环的基本用法和 for 循环的基本用法；第 7 章数组，主要讲解数组的定义及使用方法；第 8 章指针，主要介绍指针的概念、定义、使用方法及指针作为参数的地址传递方法；第 9 章函数，主要讲解函数的概念、定义及函数参数传递；第 10 章编译预处理，主要介绍常用的预处理命令，包括宏定义、文件包含等 C 程序常用预处理；第 11 章位运算，主要介绍对数据按位进行运算的方法，包括按位逻辑运算和移位运算；第 12 章结构体和共用体，主要介绍结构体的定义和使用；第 13 章文件的操作，主要介绍文件的操作方法。

本书第 1 章由张树静编写，第 2 章由赵欣编写，第 3 章由张海燕编写，第 4、5 章由赵晓莉编写，第 6、7 章由胡鹏飞编写，第 8、9 章由张丽君编写，第 10、11、12 章由魏红娟编写，第 13 章由李琳芳编写。

本书在编写过程中借鉴了部分经典教材的内容，在此向相关作者表示感谢。

尽管作者根据多年的教学经验，试图让初学者用最少的时间来学会 C 语言程序的开发，但由于受水平的限制，书中难免会有不足之处，欢迎广大读者批评和指正。

编　者

2013 年 12 月

目　录

第1章 概 述

1.1 历史背景

世界上第一台计算机诞生于1946年的美国，随着时间的推移，计算机的应用从国防和科学计算，发展到日常家庭生活。计算机的英文computer，是由动词compute(计算)加上er后缀得到的，指用于计算的机器。众所周知，计算机所能识别并采用的是机器语言，也就是二进制语言，由0和1组成，如果要求计算机做某些计算，就需要用二进制语言来编写一系列的指令，然后输入计算机，计算机根据这些指令序列进行特定的计算和判断。最早期的计算机语言，常常为了完成一个极其简单的任务或者计算，不得不编写一大串的指令，这项工作无论对编程者还是读程序的人来说，都是一件极其不容易的事情。虽然后面也出现了一些其他的计算机语言，如汇编语言，但是相对来说，其过程还是繁琐的，无法满足人们日益增长的程序开发的需要。为了加快程序开发的进程，人们逐渐设计出一些可以由一条语句就能够完成一个特定任务的高级语言，如C语言、C++、Java等。在这些语言中，C语言可以说是功能最强大、使用最广泛的语言之一。

现在的计算机能够以比人类快几百万甚至几十亿倍的速度来进行计算和逻辑上的判断，这一点是如何实现的呢？事实上，计算机在进行庞大的数据处理时，就是在一系列的指令集下按照程序设计的流程来进行一些列的操作，这些指令的集合，就称之为计算机程序，也就是计算机语言。计算机是由计算机程序来控制的，而人则负责编写这些计算机程序，从而使得计算机最终按照人的意思进行一系列的数据处理和分析判断。C语言是一门计算机语言，更是一个关于程序设计的方法的集合。

在了解C语言之前，我们先来了解一下它的历史。C语言并不是突兀地诞生的，它是从B语言演化而来的，而B语言则起源于BCPL语言。B语言是Ken Thompson在模拟并且继承了BCPL语言的诸多特点的基础上，经过了进一步的简化，从而开发出来的。B语言诞生之后，它过于简单，所以在应用过程中，它在很多方面的功能受到了限制。在1972到1973年间，贝尔实验室的Dennis Ritchie将B语言进行了进一步的优化，从而设计出C语言，C语言的命名取自BCPL的第二个字母。1978年，Brian Kernighan和Dennis Ritchie出版了《The C Programming Language》，此书被认定为C语言的非正式的标准说明，从此开始，贝尔实验室正式发布了C语言。后来到了20世纪七八十年代，大型主机和小型微机都广泛地应用了C语言，从而也衍生了不同版本的C语言。1983年，美国国家标准局(ANSI，American National Standards Institute)成立了一个委员会，专门来制定C语言的标准。后来，随着C语言的发展和越来越广泛的应用，C语言标准也经过了多次的修订和增补，至2011年12月8日，ISO正式公布C语言新的国际标准草案：ISO/IEC 9899:2011，即C11。

简而言之，C 语言从诞生到现在，取得了举世瞩目的成就。C 语言不仅仅保留了 B 语言的很多优点，又克服了 B 语言过于简单的缺点，增加了数据类型以及其他一些强大的功能，并且采用了结构化的程序设计思想，使得它的结构极其严谨，从而便于把由 C 语言编写的程序移植到大多数计算机上，因此得到了全世界的认可和欢迎，C 语言也因此成为世界上应用最广泛的几种计算机语言之一。

虽然人们此后在 C 语言的基础上又开发出了 C++、Java 以及 C# 语言等，但是 C 语言是它们的基础。计算机语言的发展如同一棵大树的成长过程，机器语言、汇编语言是计算机语言的根，C 语言是大树的躯干，而其他的高级语言则如同大树的枝叶，只有掌握了基础和躯干，才能以不变应万变，立于不败之地。

我们可以先简单了解一下由 C 语言衍生出来的几个主要的程序语言。

C++：C++ 语言与 C 语言有着很深的渊源，C++ 语言基本兼容了 C 语言的全部语法，当今所流行的 C++ 编译器都可以编译 C 语言，以至于 C++ 语言被误认为是 C 语言的升级版。但是事实上，C++ 语言采用面向对象的程序设计思想，而 C 语言采用的是结构化的程序设计思想，二者程序设计的思维方式是完全不同的。C++ 里面的“++”表示“Object Oriented”(面向对象)。通俗的说法是，C 语言更贴近低级语言，而 C++ 则更符合人类语言的表达习惯。

Java：Java 语言诞生于 1995 年，是由 Sun 公司开发出来的，纯面向对象、与平台无关且易学易用。它全面照搬了 C++ 的语法，并且去掉 C++ 里面不常用和不成功的部分，做到化繁为简，从而获得越来越多用户的支持。

C#：C# 语言是由微软公司于 2000 年推出的，完全照搬了 C/C++ 的语法特点，同时又吸收了 Java 的成功之处，从而变成了一个似 Java 非 Java，在某些地方又胜于 Java 的语言。

在程序语言的世界里，C/C++、Java、C#呈三足鼎立之势，并且都在不断完善、不断发展且极力排它，以至于其他语言只能在其专属领域里发展。C 语言是一种高级语言，但是它同时还具备了一些低级语言的特点，也有人戏称 C 语言为“中级语言”。作为一种比较广泛流行的热门语言，C 语言有着其不可忽视的优点和特点。通过本书的学习，读者将了解到 C 语言的特性，并且能够读懂并且编译一些简单的 C 程序，为以后程序语言的学习打下良好的基础。

1.2 C 语言的特点

C 语言能够得到如此广泛的应用，必然是有着其他程序语言所不可比拟的优点。C 语言的主要特点如下：

(1) 表达方式灵活。它只规定了 32 个关键字和 9 种控制语句，C 语言所使用的 9 种流程控制语句，每一种都可以完成一个特定的任务，并且书写的形式也相对自由。对于编写程序和读程序的人来说，都显得简洁明快，一目了然。

(2) 数据类型丰富。与机器语言相比，C 语言增加了数据类型，定义了整型、实型、字符型等数据类型，又引入了指针的概念。这些特点足以实现各种复杂的数据运算和逻辑判断，使程序的运行效率更高。

(3) 运算符丰富。C 语言不仅仅采用了数学里面常用的加减乘除等运算符，还把括号、问号、逗号等符号引用到运算符里面去，从而扩大了其运算的类型，也使得各种运算符和各种数据类型组成的表达式呈现多样化，可以实现难度较大的运算。这是其他的高级语言不可比拟的。

(4) 结构化的语言。结构化的语言的特点在于代码及数据的分隔化，也就是说，程序的各个部分除了必要的数据交流外，彼此之间是相互独立的。因为 C 程序语言把函数作为其基础单元，每个函数都可以进行单独的设计，也就是实现了程序的模块化设计，这样使得编程更自由灵活，同时程序的层次也更清晰，便于使用、维护和调试。

(5) 可移植性好。因为 C 语言可以实现汇编的一些优点，也就是能够对硬件进行操作，例如，C 语言可以对位、字节和地址进行操作，这是低级语言的特点；同时其可以编写系统软件，又可用来开发应用软件，已成为一种通用的程序设计语言；同时 C 语言适合于多种操作系统，如 Windows、DOS、UNIX 等，支持多种显示屏和驱动器。这些显著的优点，使得它的适用范围远远超过了其他语言。

(6) 语法限制不太严格。程序设计的自由度是由语法限制来决定的，如果语法限制得太严格了，那么自由度就小了，也就失去了灵活性，但是如果完全强调灵活，那么程序则变得毫无章法，没有了意义。C 语言的语法限制相对宽松，如对整型量与字符型数据及逻辑型数据可以通用等，就是给了程序员相对较大的自由度，由程序员在编写的时候自己控制，但是对程序员的熟练度的要求就更高了。

(7) 生成目标代码高。与汇编语言相比，C 语言描述问题的工作量小并且可读性高，更易于调试、修改和移植，所生成的代码质量却与汇编语言相当。一般情况下，C 语言只比汇编语言生成的目标代码低 10%～20%。也就是处理同样的问题，C 语言更简洁易行一些。

C 语言的缺点主要表现在数据的封装性上，这是 C 语言数据安全性上所存在的最大的缺陷。同时由于 C 语言的运算法过于丰富，从而造成优先级太多；以及 C 语言的语法限制不太严格，书写过于自由，可读性下降；还有对变量的类型约束也不严格，从而影响了程序的安全性。此外，C 语言对算法正确与否和数组下标越界都不做检查等，使得 C 语言在编辑、编译和运行中产生了一些不必要的错误。从应用的角度来看，C 语言确实比其他的高级语言更难掌握一些。因此，对 C 语言的运用者来说，C 语言的这些特点也就是要求他们对程序设计更加熟练一些。

1.3　简单的 C 程序

下面介绍两个简单的 C 程序，然后了解一下 C 语言的特点。

【例 1-1】 一个只包含输出语句的小程序。

```
#include<stdio.h>                    /* 预处理命令 */
main()                               /* 主函数 */
{
    printf("O.K!\n");                /* 输出语句 */
}
```

程序运行结果如图 1.1 所示。

图 1.1　例 1-1 运行结果

这是一个完整的 C 程序。

第一行是预处理命令 #include，用来调用函数库里面的标准函数。C 语言中不存在输入/输出语句，若要用到输入/输出，就需要预先调用 C 语言函数库里的输入/输出库函数，stdio 是标准输入/输出(standard input&output)的英文缩写，后面的 .h 是 head 的缩写，.h 是头文件的扩展名。

第二行是主函数 main。在 C 程序中，至少有一个 main 函数，main 函数中包含一个小括号。

第三行至第五行是函数体，由大花括号 {} 括起来，函数体里只有一个输出语句，由 printf()函数和分号组成。printf()就是预处理命令所调用的输入/输出库函数里面的输出函数；而 C 语言的每一个语句都以分号作为语句的结束，这条语句的作用就是输出 printf() 的小括号里面的引号里的内容。在引号里面，还有一个“\n”，它是一个换行符，意义就是在输出 O.K! 之后回车换行。

【例 1-2】 求两个数之差的绝对值。

```
#include<stdio.h>                    /* 预处理命令 */
main()
{
   int x,y,z;
   scanf("%d，%d",&x,&y);            /* 输入函数语句 */
   z=tbs(x-y);                       /* 计算 x－y 的值并取绝对值后赋给变量 z */
   printf("z=%d\n",z);               /* 输出函数语句 */
}
int tbs(int a)                       /* 变量声明 */
{
    int b;
    if(a>=0)b=a;
    else b=-a;
    return(b);
}
```

例 1-2 的 C 程序包含了两个函数：一个主函数 main()，一个自定义函数 tbs()。tbs() 函数的意义是对数值进行取绝对值运算。

程序的前两行是预处理命令，调用标准输入/输出库函数。

第三行是主函数 main() 函数。

第四行定义了三个整型变量：x, y, z。

第五行是输入函数语句，输入函数 scanf() 的作用是要求以十进制整数的形式给变量 x 和 y 赋值，也就是从键盘输入两个整数分别送到变量 x 和 y 的存储单元中去。“&”是取地

址运算符，它的作用是得到变量 x 和 y 的存储单元的首地址，这些知识点我们会在以后的章节里陆续学到。

第六行是调用 tbs() 函数，在调用时，将 (x – y) 的值传给 tbs() 函数里的形式参量 a，经过执行 tbs() 函数，得到一个返回值，最终赋给变量 z。

第七行是输出程序运行的结果，printf() 中引号内的普通字符“z=”原样输出，“%d”是要求在输出 z 的值时以十进制整数的形式输出。

第八行至第十三行是 tbs() 函数的具体内容。

下面来看一下程序运行的情况：

```
3,5 ↙        (输入 3, 5，按回车读入，分别赋给 x, y)
```

程序运行结果如图 1.2 所示。

图 1.2　例 1-2 运行结果

例 1-2 程序比例 1-1 程序复杂了一点，用到了数据类型中的整型变量、实际参数、形式参数等，这些内容将在以后的章节中详细讲解，我们就不再做进一步的解释了。通过这两个例子，读者应该对 C 语言程序的组成和形式有了一个初步的认识。下面我们总结一下 C 语言程序的基本结构。

(1) C 源程序中包含头文件。头文件可以是对程序中所用变量的说明，也可以是引用的库函数。我们统一称之为预处理命令；如上面两个例子中的所用的 include 命令。除此之外，头文件还有可能包含 ifdef 命令、define 命令等。预处理命令一般放在源文件或者源程序的最前面。

(2) C 程序是由函数构成的。每一个 C 语言源程序可以由一个或多个源文件组成，而每一个源文件则由一个或者多个函数组成。但是不管一个源程序由多少个文件组成，这个源程序里面有且只有一个 main() 函数，即主函数。而 C 程序总是从 main() 函数开始整个程序的执行，并且终止于主函数 main()，与主函数 main() 所在程序中的位置无关。除了 main() 函数之外，C 程序还可以包含零个到多个用户自定义的函数，但是 main() 函数有且只有一个。函数是 C 语言的最基本的组织结构单元。

(3) 函数由两部分组成。每一个函数都是由两部分组成的，即函数的首部和函数体。在例 1-2 里，int tbs(int a) 就是函数的首部，它包含了函数名(tbs())、函数类型(int)、函数参数名(a)、函数参数类型(int)，这些都是对函数的返回值类型、函数名及参数类型的定义和说明。当然，函数也可以没有形式参数，但是函数名后面的小括号是不能省略的(如 main() 函数)。函数体指的是紧跟在函数首部下面的最外层的大花括号里面的内容。函数体还可以分成两部分：变量的定义和执行语句。在例 1-2 里面，main()函数的函数体是从第四行到第八行的部分，由一个大花括号括起来。在这个函数体里面：第一行是变量的声明语句，后面三行则可以统称为执行语句。变量的声明是对函数中所要使用的变量进行数据类型的定义，在这个例子里面，是将变量 x 与 y 定义为整型数据。而执行语句则是函数算法的体现，也就是函数要实行的“动作”的描述，这也是这个程序的意义所在。

除此之外，还有几点需要特别注意的书写规则：

(1) C 语言的每一个语句都是由表达式加上分号构成。只有表达式而无分号，则不是一条语句，分号是一条语句的结束标志。但是在预处理命令、函数头和大花括号“}”之后不能加分号。(结构体、联合体、枚举型的声明的大花括号“}”之后要加“;”。)

(2) C 语言的书写是比较自由的。可以一个语句写一行，也可以多条语句写在一行，甚至还可以将一条语句写成多行(在流程控制语句中体现)，且不需要加任何标识符。但是为了程序的可读性和可执行性，建议读者养成一行内只写一条语句的习惯，这样比较规范，也便于阅读和纠错。

(3) 标识符、关键字之间一般加一个空格以示间隔。如果已经有了明显的间隔符，则也可以不加空格。

(4) 每一行都对应相应的注释部分。注释部分放在/*.....*/内，可以用自然语言来书写，一行写不完，可以写到下一行。注释部分是为了提高程序的可读性，便于程序员检查和纠错。

(5) 低层次的语句最好比高层次的语句缩进若干格后书写，这样看起来结构更加清晰，也增加了程序的可读性，同时方便程序员在编译中纠错。

通过上面的例子，我们简单地了解了 C 程序的基本结构和 C 程序的组成单元函数，同时还有一些书写方面的特征，这些仅仅是 C 程序的入门知识，在以后的章节里面我们还将具体到每个细节，更详细地介绍其相关要点。

1.4 C 语言字符集、标识符与关键字

1.4.1 C 语言字符集

C 程序作为一种计算机语言，必定有其所使用的固定字符，并且所使用的字符还会受到硬件设备的限制。要编写一个能够在计算机上运行的 C 程序，就只能使用符合 C 语言规定的合法的字符。C 语言所使用的基本字符可分为六大类：运算符、分隔符、常量、注释符、标识符、关键字等。组成基本字符的字符集包括英文字母(大小写字母共 52 个)、阿拉伯数字(0～9)及其他一些特殊符号(下划线等)。特殊符号如下所示：

```
+  -  *  /  %  ++  ——
<  >  =  >=  <=  ==  !=
!  ||  &&  ,  ?  :  ;  .  \
()  []  {  }  &  '  "  ^
~  |  <<  >>
```

1.4.2 C 语言标识符与关键字

1. 标识符

C 语言中的标识符是指用来表示函数、类型以及变量的名称的字符，是编程者自己为函数、类型、变量所起的名称，一般由数字、字母和下划线组成，并且开头必须是字母或

者下划线，后面可以跟若干个(或者 0 个)字母、数字和下划线。

由于在 C 语言里面，大小写是有区别的，所以在定义标识符时，大小写不同的则是不同的标识符，如 SUM 与 sum 是两个不同的标识符。并且在定义标识符时，每个标识符最好有相应的意义，以便读者可以“顾名思义”，从而增加程序的可读性。还有需要注意的一点是标识符的长度，虽然标准 C 不限制标识符的长度，但是由于存在各种版本的 C 语言编译系统的限制，标识符一般不要超过 8 个字符。所以在定义标识符时，标识符的字符个数不要取得太长，以免出现不必要的错误。

下面给出一些合法的标识符：

AB13　　EF_3　　XY4_　　_DC6

下面是一些不合法的标识符：

4CG　　B.C.Jon　　#A73　　if

第一个不合法的标识符是以数字开头的，这不符合标识符的命名规则，同样的道理，第二个运用了“.”这个符号，第三个则用到了“#”，这些都不属于标识符命名所能采用的符号，因为标识符命名只限定是字母、数字和下划线，其他的特殊符号一概不能用到标识符里面去。最后一个是关键字，关键字是不能重新命名为标识符的，原因在下面会讲解的比较清楚。

2. 关键字

关键字是 C 语言中已经规定了有其特殊意义的标识符，不允许用户再进行自定义使用。也就是说，用户只可以根据规定使用，而不可以使用其作为函数名或变量名。关键字都采用小写字母。C 语言中有 32 个关键字，如下所示：

if　else　for　do　while　goto　int　long　short　char　auto　break　const
continue　default　case　double　enum　extern　float　register　return　signed
sizeof　static　struct　switch　typedef　union　unsigned　volatile

1.5　C 程序的开发环境

当一个 C 程序写好之后，就要将其放到编译系统里面进行编辑(输入)、编译和连接，只有在完成了上面一系列步骤之后，一个 C 程序才变成一个可执行的程序，然后在计算机上运行，从而得到结果。C 语言程序的上机过程如图 1.3 所示。

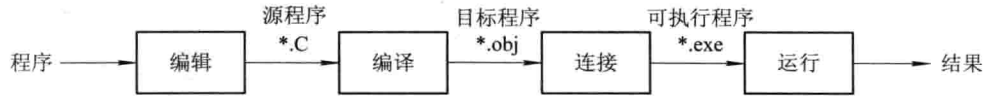

图 1.3　C 语言的上机过程

当前有很多种编译系统可以对 C 语言程序进行编译，本书着重介绍 Visual C++6.0 语言集成环境，读者可以了解 C 语言程序在此环境下是如何进行编辑、编译、连接和运行的。同时还将简单地介绍一下 TurBo C2.0 集成环境，它是 DOS 环境下的一款 C 语言集成环境，作为了解即可。

1.5.1 Visual C++6.0 简介

Visual C++ 6.0，简称 VC 或 VC6.0，是微软推出的一款将高级语言翻译为机器语言的，并且是针对 C++ 面向对象程序设计语言的开发环境。由于 C++ 语言是在 C 语言的基础上扩展而成的，所以 C 语言也能在 VC6.0 上运行。VC6.0 不仅仅是一个 C++ 编译器，而且是一个基于 Windows 操作系统的可视化集成开发环境。VC6.0 是一个功能强大的可视化软件开发工具，以其拥有“语法高亮”、自动编译功能和高级除错功能而著称。在 VC6.0 上对 C 程序进行编辑和编译时，由于它允许用户进行远程调试、单步执行，并且允许用户在调试期间重新编译被修改的代码，而不必重新启动正在调试的程序，再加上其编译及创建预编译头文件、最小重建功能及累加连接等特点，从而大大缩短了程序编辑、编译及连接所花费的时间，使其成为专业程序员软件开发的首选工具。

了解了 VC6.0 的这些特点之后，下面来看如何在 VC6.0 环境下编辑、编译及运行一个 C 程序。

1. 启动 VC6.0

单击“开始|程序|Microsoft Visual studio|Microsoft Visual C++6.0”，启动 VC6.0，在电脑屏幕上将显示如图 1.4 所示的窗口。

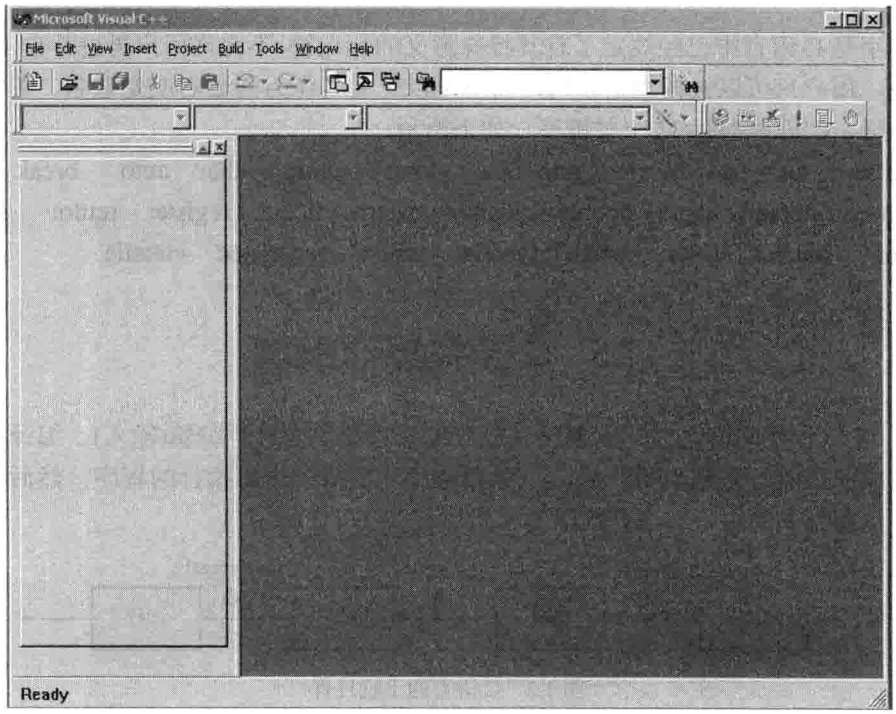

图 1.4　VC6.0 窗口

在此窗口的上面，我们可以看到有菜单项，单击每一个菜单项，都有相应的下拉菜单，在下拉菜单里面可以执行相应的操作。如在图 1.5 和图 1.6 所示的 File 的下拉菜单中，如果要新建一个源程序，就可以选择 New 选项来新建源文件。

图 1.5　菜单栏

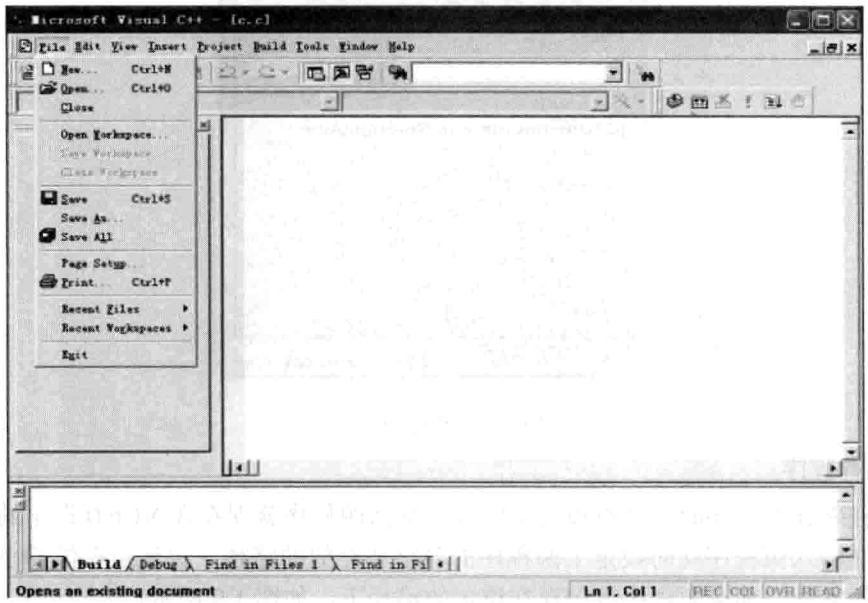

图 1.6　新建源文件窗口

2. 新建 C 源程序

单击 VC6.0 窗口上的“文件|New”命令，弹出“New”对话框，如图 1.7 所示。单击对话框中的“Files”选项卡，单击“C++Source File”，然后单击“OK”按钮，就进入了编辑窗口，输入源程序，这时候源程序名及路径都是系统给的默认值。

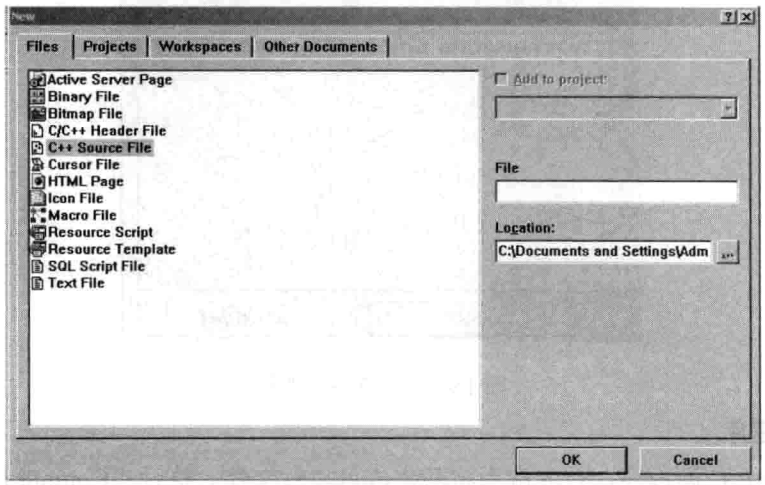

图 1.7　系统默认的程序名